

# Algorithm of computing verbal relationships for generating Vietnamese paragraph of summarization from the logical expression of discourse representation structure

Trung Tran<sup>1</sup> · Dang Tuan Nguyen<sup>1</sup>

Received: 14 May 2015 / Accepted: 23 November 2015 / Published online: 30 December 2015  
© The Author(s) 2015. This article is published with open access at Springerlink.com

**Abstract** This paper presents an important phase of our new approach for summarizing the given Vietnamese paragraph. The central of this phase is an algorithm for computing verbal relationships in the process of generating the Vietnamese paragraph from the logical expression of discourse representation structure (DRS), which is the first-order logic expressions without explicit quantifiers, and represents the meaning as well as reflects the potential contexts of a given discourse or a sequence of sentences. By defining elements to describe the appropriate information in each predicate of the logical expression (or can be called “DRS-conditions”), the algorithm is based on in turn considering three consecutive predicates in a logical expression for determining: the relationship between the first and second sentence, the relationship between the second and third sentence, and the priority when comparing these two relationships. The evaluation achieves two given criteria: the semantic completeness of summarization, and the natural quality of new reduced paragraph.

**Keywords** Logical expression · Discourse representation structure · Sentence generation · Predicate relationship

## 1 Introduction

In general, the study of transforming a given paragraph to a new summary (Das and Martins [7], Lloret [20],

Mani and Mayburi [21], Jezek and Steinberger [13], Jones [14,15]) has to answer three important questions (Jones [14,15]): (i) how to represent the meaning of the source paragraph; (ii) how to construct a computing representation form of the destination paragraph by transforming the source computing representation form; (iii) how to transform the destination computing representation form into the complete paragraph. These lead to two main approach directions: (i) extract some sentences which have highest benchmarks to produce the summary—this direction is called “extraction”; (ii) construct a summary based on understanding the meaning of the source paragraph—this direction is called “abstraction”.

Follow the idea of “abstraction”, this paper addresses an important problem in our new approach for summarizing the given Vietnamese paragraph having more than two simple sentences: generate the new reduced Vietnamese paragraph from the logical expression of discourse representation structure (DRS) (Kamp [16], Covington and Schmitz [5], Covington et al. [6], Blackburn and Bos [1])—under the form of expressions without explicit quantifiers in first-order logic (FOL). To limit the scope of this article, we assume that there were the methods for mapping the original paragraphs to the logical expressions encodings of their meanings (Zettlemoyer and Collins [34,35]). Using the logical expression representing the semantic of the paragraph, our objective is to propose a solution for transforming this logical expression into a new reduced complete Vietnamese paragraph.

As an example, consider the following original Vietnamese paragraph consisting of four simple sentences:

*Example 1 “Lan vui vẻ. Cô học cùng con trai. Nó khoẻ chỉ. Nó được điểm cao.”*

✉ Trung Tran  
ttrung@nlke-group.net  
Dang Tuan Nguyen  
dangnt@uit.edu.vn

<sup>1</sup> Faculty of Computer Science, University of Information Technology, VNU-HCM, Ho Chi Minh City, Vietnam

(English: “Lan is happy. She studies with son. He is overjoyed. He takes high mark.”)

The logical expression representing the semantic of paragraph in Example 1 is illustrated in two forms:

- DRS form:

---

$[x, y, z]$
lan (x)
vui_vẻ (x)
con_trai (y)
học_cùng (x, y)
khoái_chí (y)
điểm_cao (z)
đủ'ợ'c (y, z)

---

- FOL form:

$\exists x \exists y \exists z [\text{lan}(x) \& \text{vui\_vẻ}(x) \& \text{con\_trai}(y) \& \text{học\_cùng}(x, y) \& \text{khoái\_chí}(y) \& \text{điểm\_cao}(z) \& \text{đủ'ợ'c}(y, z)]$ .

With this logical expression, we transform into the new reduced Vietnamese paragraph as follows:

“Lan vui vẻ vì học cùng con trai. Con trai khoái chí vì được điểm cao.”

(English: “Lan is happy because of studying with son. The son is overjoyed because of taking high mark.”)

In the logical expression, in which two representation forms are equivalent, the DRS form represents the meaning as well as reflects the context change potential of the given discourse. The FOL form is used for expressing the semantic for DRS form. In the above case, the logical expression represents:

- Instances “lan”, “con trai” (the son) and “điểm cao” (high mark) by predicates which associated with variables  $x$ ,  $y$  and  $z$ :  $\text{lan}(x)$ ,  $\text{con\_trai}(y)$ ,  $\text{điểm\_cao}(z)$ .
- Actions and states of these instances through predicates which associated with variables  $x$ ,  $y$ ,  $z$  appropriately:  $\text{vui\_vẻ}(x)$ ,  $\text{học\_cùng}(x, y)$ ,  $\text{khoái\_chí}(y)$ ,  $\text{đủ'ợ'c}(y, z)$ .

The considered objects in this research are complete logical expressions encoding the semantics of Vietnamese paragraphs.

The heart of the proposed solution is an algorithm that auto generates the new reduced Vietnamese paragraph from the logical expression. With the given requirement that the generated paragraph has to satisfy the universality in common Vietnamese communication, our algorithm in turn considers three consecutive predicates representing actions

and states of instances. The algorithm compares two predicate pairs [ $\text{predicate}(1)$ ,  $\text{predicate}(2)$ ] and [ $\text{predicate}(2)$ ,  $\text{predicate}(3)$ ] in considered three predicates based on the sustainable priority factor which proposed by us. The pair having the higher priority will be used for generating the syntactic structure of a new Vietnamese sentence, then combined with lexicons for completing. The remaining predicate is handled in two ways: re-create the original Vietnamese sentence or combine consideration with next two predicates in the logical expression. The algorithm is performed based on our assumption that a paragraph has the natural quality if each sentence in this has the natural quality.

To evaluate the effectiveness of the generating solution in this study, we establish two criteria: (i) the first criteria is the semantic completeness—in the sense of the generated paragraph has the content which correctly summarizes the meaning of the source paragraph; (ii) the second criteria is the natural quality—in the sense of each sentence in the generated paragraph has the native form of Vietnamese usage.

The organization of this paper is as follows. Section 2 provides a literature review of abstraction summarization direction. Section 3 presents an overview of our works with the new approach based on abstraction direction. The main content in Sect. 4 is about the heart of our solution which is the algorithm of computing verbal relationships for generating the new reduced Vietnamese paragraph. Next, in Sect. 5, we describe the experiment and indicate some analysis according to the results. Finally, Sect. 6 concludes this paper and presents future research directions.

## 2 Abstraction summarization literature review

Generally, the methods in abstraction direction can be classified into two categories (Kasture et al. [17], Khan and Salim [18], Saranyamol and Sindhu [25]): (i) structured based in which the researcher tried to determine the most important content using some structures such as tree, template, ontology, ...; (ii) semantic based in which the authors introduced some methods in natural language generation (NLG) to make the semantic representation.

### 2.1 Structured-based approach

#### 2.1.1 Tree-based method

Researchers following this method illustrated the content of given document using a dependency tree. This method is often applied for summarizing multi-documents.

Barzilay et al. [2] proposed a solution in which firstly they preprocessed the similar sentences in some news articles. Then, a theme intersection algorithm had been used for

determining the common phrases which will be transferred to FUF/SURGE language generator so that the new summary sentences were created. Although using a language generator help for reducing repetitions and increasing fluency, there was lacking in the context in which similar sentences in different document appeared while determining the intersected phrase.

In another research, Barzilay and McKeown [3] worked on sentence fusion by integrating information in overlapping sentences. Firstly, they analyzed the sentences and illustrated by the dependency trees. They determined the centroid of these trees to build a main tree and then augmented with the sub-trees of other sentences. The main drawback of this approach is that a complete model to present the abstract representation of selected content was not proposed.

### 2.1.2 Template-based method

In this approach, the researcher built a template which is text snippets to represent the given documents and generate the summary. They applied rules in an Information Extraction system (Harabagiu and Lacatusu [12]) to extract information from multiple documents. This information was used to fill the template and then generate coherent, informative multi-document summaries. The limitation of this approach is that it requires the summary sentences that are already present in the source documents and cannot identify the similar and different information across multiple documents.

### 2.1.3 Ontology-based method

Applying ontology, especially fuzzy ontology, to improve the process of summarization is one of the most interesting method. This helps for handling the uncertain data and well summarizing documents on websites which have own knowledge structure. However, because the domain experts had to make a lot of effort to define dictionary and news corpus, up to now, this approach is limited to Chinese news (Lee et al. [19]).

### 2.1.4 Lead and body phrase method

Studies in this approach focused on rewriting the lead sentence using inserting and substituting phrases in the lead and body sentences which have same syntactic head chunk. Tanaka et al. [26] proposed a method following this approach in broadcast news. They determined the maximum phrases of each same chunk in the lead and body sentences. The substitution and insertion operations were applied to these phrase in order to revise the lead sentence. By using this method, they could find

the semantically appropriate revisions. However, similar to other structure-based methods, it is the lack of a complete model.

### 2.1.5 Rule-based method

Genest and Lapalme [8] presented a method with three main modules: (i) information extraction determined several candidate rules for each aspect of verbs and nouns; (ii) content selection selected the best rule for each aspect; (iii) summary generation formed the output text using generation patterns. With this method, the researchers created summaries with greater information. On the other hand, they had to make a lot of effort to manually write all the rules and patterns.

## 2.2 Semantic-based approach

### 2.2.1 Multimodal semantic model

A framework was proposed by Greenbacker [10] for generating abstractive summary with three main steps: (i) they used ontology to build a semantic model representing the contents of multimodal documents; (ii) the metric rated the concepts in ontology with several factors such as the completeness of attributes, the number of relationships with other concepts, ...; (iii) the generator built the summary with the most important concepts. The idea of producing the abstract summary is the most important distribution of this framework, because it includes salient textual and graphical content. One point that needs to be deeply researched is that the evaluation for this framework is manually handled by human.

### 2.2.2 Information item-based method

Another research in multi-document abstraction summarization worked by Genest and Lapalme [9] focuses on generating the summary from the abstract representation of original documents called information item. They introduced a framework for summarizing with main modules: (i) information item retrieval module parsed source text and extracted subjects of verb and objects; (ii) sentence generation module creates a new sentence; (iii) sentence selection module evaluated the generated sentences generated with appropriate score; (iv) summary generation module combined highly scored generated sentences with information about dates and location to construct the whole summary.

Although the summary is short, coherent, information rich and less redundant, there are some limitations in this method: some information items which are difficult for creating meaningful and grammatical sentences can be eliminated; in information item retrieval module, if the parser could not

parse correctly the syntactic tree, then the linguistic quality of summaries is low.

### 2.2.3 Semantic graph-based method

Moawad and Aref [22] constructed a semantic graph called rich semantic graph to represent the semantic of source document. This graph was then reduced using some heuristic rules and transformed to the abstractive summary. The output summary of this method could be concise, coherent and less redundant. However, this method lacked of knowledge about linguistic theories, then the summary may be not grammatically and naturally correct in applied languages.

## 3 Overview of paragraph summarization by generating reduced paragraph

Follow the idea of “abstraction” but with the approach which is to combine the knowledge and techniques in text understanding and representing, text generation (Reiter and Dale [23,24]) as well as functional grammar linguistic theory (Cao [4], Halliday and Matthiessen [11]), we proposed in [30] a specification model called Verbal Relationship-based Computational Model (VRBCM) to formalize the main idea of our summarization solution. This model consists of four main components: The first three sets help for specifying understanding the meaning of the original paragraph—set of lexical information representations, set of inner relationships of each sentence, set of inter-sentential relationships between each pair of consecutive sentences; the last set helps for specifying generating the new paragraph—this set contains syntactic structures of sentences of the summary. The foundation of this model is the hypotheses about four types of inter-sentential relationships between each pair of consecutive sentences in the original paragraph: objective, cause, consequence, concurrence.

Implementing and applying model VRBCM, focusing on the phase of transforming the source representation form to summary, we based on considering objective and consequence inter-relationships between two sentences to propose in [27–29,33] methods and techniques to summarize some pair types of Vietnamese sentences having suitable characteristics.

At the phase of understanding the meaning of the source pair of Vietnamese sentences, we proposed in [31,32] strategies for resolving the ambiguity when considering inter-anaphoric pronouns appearing in some pair types of Vietnamese sentences having special characteristics.

## 4 Generation of summarizing paragraphs

In this section, we present the heart of our solution which is the algorithm of computing verbal relationships for generating the new reduced Vietnamese paragraph. The input of the algorithm is predicates representing actions or states in the logical expression. The main idea of the algorithm is to consider, in turn, three consecutive predicates, determine the pair of predicates having the higher relationship priority and generate the syntactic structure of the new Vietnamese sentence based on the relationship of this pair. Thus, at a high level, the algorithm will involve the following three sub-problems:

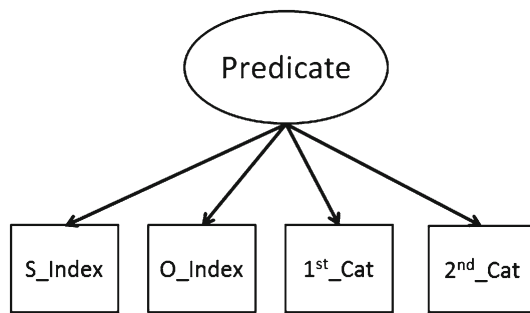
- Determine predicates representing actions or states.
- Generate the syntactic structure of the new Vietnamese sentence based on the relationship of one pair of predicates.
- Determine the relationship priority in comparison between two pairs of predicates.

In the remainder of this section, we describe an overall strategy for these three problems. Section 4.1 presents the characteristic structure of one predicate which is defined for this research and the algorithm for selecting predicates representing actions or states. In Sect. 4.2, we synthesize relationship types between two predicates representing actions or states based on considering the characteristic structure. Also in this section, corresponding to each relationship type, we present constructing the syntactic structure of the new Vietnamese sentence. Finally, in Sect. 4.3, we present handling the third problem and describe in general the algorithm for generating the new Vietnamese paragraph.

### 4.1 Predicate characteristic structure

In this research, we limit the consideration of action or state sentences. The verbs indicating actions or states belong to one of the four categories with meanings (based on the categorization in theory functional grammar [4,11]):

- The first category is called action “intransitive”. The verbs belonging to this category indicate an action which associates to only one actor.
- The second category is called action “transitive”. The verbs belonging to this category indicate an action which associates to one actor and one goal.
- The third category is called state “status”. The verbs belonging to this category indicate existing temporary status of a subject.
- The forth category is called state “property”. The verbs belonging to this category indicate a property inside a subject.



**Fig. 1** The characteristic structure of a predicate in the logical expression

Based on the above categorization, we define the characteristic structure of a predicate in the logical expression composing components in Fig. 1:

In this structure, each component takes value as follows:

- Component *S\_Index* taking the value as an index (represented by one bound variable) indicates the instance taking the subject role.
- Component *O\_Index* taking the value as an index (represented by one bound variable) indicates the instance taking the object role.
- Component *1st\_Cat* taking the value as an index (represented by one bound variable) indicates the category at the first level: object/action/state.
- Component *2nd\_Cat* taking the value as an index (represented by one bound variable) indicates the category at the second level: proper/common/intransitive/transitive/status/property.

As an example, consider the logical expression in Sect. 1. The predicates in this expression have the characteristic structure with components taking values as follows:

lan	$:= \{S\_Index \rightarrow x; O\_Index; 1st\_Cat \rightarrow object; 2nd\_Cat \rightarrow proper\}$
vui_vẻ	$:= \{S\_Index \rightarrow x; O\_Index; 1st\_Cat \rightarrow state; 2nd\_Cat \rightarrow status\}$
con_trai	$:= \{S\_Index \rightarrow y; O\_Index; 1st\_Cat \rightarrow object; 2nd\_Cat \rightarrow common\}$
học_cùng	$:= \{S\_Index \rightarrow x; O\_Index \rightarrow y; 1st\_Cat \rightarrow action; 2nd\_Cat \rightarrow transitive\}$
khoái_chí	$:= \{S\_Index \rightarrow y; O\_Index; 1st\_Cat \rightarrow state; 2nd\_Cat \rightarrow status\}$
điểm_cao	$:= \{S\_Index \rightarrow z; O\_Index; 1st\_Cat \rightarrow object; 2nd\_Cat \rightarrow common\}$
được	$:= \{S\_Index \rightarrow y; O\_Index \rightarrow z; 1st\_Cat \rightarrow action; 2nd\_Cat \rightarrow transitive\}$

We classify into two lists: *O\_List* consists of predicates representing instances, *AS\_List* consists of predicates representing actions or states. The main idea of this classification is based on the value of component *1st\_Cat* in each predicate. The classification algorithm:

**Algorithm 1:** Classify predicates representing instances and predicates representing actions or states.

**Input:** *P\_List* = List of predicates in the logical expression.

**Output:** *O\_List* = List of predicate representing instances; *AS\_List* = List of predicates representing actions or states.

```

n = |P_List|;
For i = 1 to n Do
    If 1st_Cat(predicate(i)) == "object"
    Then
        Put predicate(i) into O_List;
    Else Then
        Put predicate(i) into AS_List;
    End If
End For

```

Apply Algorithm 1 for predicates in the logical expression in Fig. 1, we obtain the result with two lists *O\_List* and *AS\_List*:

- *O\_List*: lan, con\_trai, điểm\_cao.
- *AS\_List*: vui\_vẻ, học\_cùng, khoái\_chí, được.

## 4.2 Predicate relationships and sentence structure generation

The main content of this section is to present establishing the assumption about relationship types between two predicates representing actions or states. Therefrom, we present generated syntactic structures of new Vietnamese sentences suitable for each relationship type. The main idea for implementing is based on the verbal categorization in Sect. 4.1.

An important requirement in this study is that the generated paragraph has to satisfy the universality in common Vietnamese communication. We accept that in order to meet

this requirement, each generated Vietnamese sentence has to satisfy the universality. Generating the new sentence having this characteristic needs to be based on considering relationships in a certain context between two original sentences. In this research, our solution is to establish an order prior-



ity for considering predicates representing actions or states. Therefrom, we propose relationship types between pairs of predicates which represent relationships between original pairs of sentences.

According to categorizing verbs indicating actions or states in Sect. 4.1, we assume a considering order priority. The basis for establishing the assumption is based on the sustainable level in the context: if the sustainable level is longer, then the considering priority is lower. Concretely, the order priority of each verbal category is as follows:

- Verbs indicating state status take the highest considering priority is (1).
- Next, verbs indicating action intransitive and action transitive in turn take the considering priority are (2) and (3).
- Lastly, verbs indicating state property take the lowest considering priority is (4).

Consider each pair of predicates representing actions or states ( $Pas_i$ – $Pas_j$ ), there are four relationship types when comparing the priority of  $Pas_i$  and  $Pas_j$ :

- $Pas_i$  having priority (2) is performed so that can perform  $Pas_j$  having priority (2) or (3);
- $Pas_j$  having the lower priority takes the role as a cause of  $Pas_i$ ;
- $Pas_j$  having the higher priority takes the role as consequence of  $Pas_i$ ;
- $Pas_i$  and  $Pas_j$  occur simultaneously if have the equal priority.

In Table 1, we synthesize all cases of these four relationship types:

We generate the syntactic structure of the new reduced Vietnamese sentence for each pair of predicates representing actions or states ( $Pas_i$ – $Pas_j$ ) based on each relationship type in Table 1. The main idea for implementing consists of the following main steps:

- *Step 1* In turn determine predicates representing instances which have the relationship with each predicate  $Pas_i$  and  $Pas_j$ . The relationship here is understood that component  $S\_Index$  in the predicate indicating instance takes the value which is identical with the value of component  $S\_Index$  or  $O\_Index$  of  $Pas_i$  or  $Pas_j$ . Therefrom, construct two syntactic structures according to  $Pas_i$  and  $Pas_j$ . Each this syntactic structure is the structure of one sentence in the source paragraph and belongs to one of two form:
  - *Case 1* Component  $2nd\_Cat$  of predicate representing action or state  $Pas$  takes value “transitive”. There are two predicates representing instance which are

**Table 1** Classify relationship types for each pair of predicates

Type	Meaning	Priority of $Pas_i$	Priority of $Pas_j$
i	$Pas_i$ is performed so that can perform $Pas_j$	(2)	(2)
		(2)	(3)
ii	$Pas_j$ is the cause of $Pas_i$	(1)	(4)
		(1)	(2)
		(1)	(3)
		(2)	(4)
iii	$Pas_j$ is the consequence of $Pas_i$	(3)	(4)
		(4)	(1)
		(4)	(2)
		(4)	(3)
iv	$Pas_i$ and $Pas_j$ occur simultaneously	(2)	(1)
		(3)	(1)
		(3)	(2)
		(4)	(4)
		(1)	(1)
		(3)	(3)

$Po1(x)$  and  $Po2(y)$  which have the relationship with  $Pas$ . The structure form is:

$$Form\_1 := Po1(x) + Pas(x, y) + Po2(y)$$

- *Case 2* Component  $2nd\_Cat$  of predicate representing action or state  $Pas$  takes other values. There is one predicate representing instance which is  $Po1(x)$  which have the relationship with  $Pas$ . The structure form is:

$$Form\_2 := Po1(x) + Pas(x, y).$$

- *Step 2* Merge two syntactic structures according to  $Pas_i$  and  $Pas_j$  to construct the syntactic structure of the new reduced Vietnamese sentence. The merging rule consists of the following steps:
  - *Step 2.1* Add elements in the syntactic structure according to  $Pas_i$  into the new structure.
  - *Step 2.2* Add the relationship factor belonging to one of relationship types in Table 1 into the new structure.
  - *Step 2.3* Determine the context is active or passive voice for the syntactic structure according to  $Pas_j$ .
  - *Step 2.4* Add elements in the syntactic structure according to  $Pas_j$  into the new structure.

**Table 2** The syntactic structure of the new reduced Vietnamese sentence according to relationship type (i)

Priority of $Pas_i$	Priority of $Pas_j$	Syntactic structure
(2)	(2)	$[Po1] + [Pas_i] + \langle i \rangle + [Pas_j]$
(2)	(3)	$[Po1] + [Pas_i] + \langle i \rangle + [Pas_j] + [Po2]$ $[Po1] + [Pas_i] + \langle i \rangle + \text{"is"}[Pas_j] + \text{"by"} + [Po2]$

**Table 3** The syntactic structure of the new reduced Vietnamese sentence according to relationship type (iii)

Priority of $Pas_i$	Priority of $Pas_j$	Syntactic structure
(4)	(1)	$[Po1] + [Pas_i] + \langle iii \rangle + [Pas_j]$
(4)	(2)	$[Po1] + [Pas_i] + \langle iii \rangle + [Pas_j]$
(4)	(3)	$[Po1] + [Pas_i] + \langle iii \rangle + [Pas_j] + [Po2]$ $[Po1] + [Pas_i] + \langle iii \rangle + \text{"is"} + [Pas_j] + \text{"by"} + [Po2]$
(2)	(1)	$[Po1] + [Pas_i] + \langle iii \rangle + [Pas_j]$
(3)	(1)	$[Po1] + [Pas_i] + [Po2] + \langle iii \rangle + [Pas_j]$ $[Po1] + [Pas_i] + [Po2] + \langle iii \rangle + [Po2] + [Pas_j]$
(3)	(2)	$[Po1] + [Pas_i] + [Po2] + \langle iii \rangle + [Pas_j]$ $[Po1] + [Pas_i] + [Po2] + \langle iii \rangle + [Po2] + [Pas_j]$

Perform step 1 and step 2, we synthesize syntactic structure forms of new Vietnamese sentences according to each relationship type in Table 1:

- Relationship type (i) (Table 2)
- Relationship type (ii) (Table 8)
- Relationship type (iii) (Table 3)
- Relationship type (iv) (Table 4).

### 4.3 The Vietnamese paragraph generation algorithm

The algorithm for generating the new reduced Vietnamese paragraph takes the input as two lists: *AS\_List* contains predicates representing actions or states and *O\_List* contains predicates representing instances (described in Sect. 4.1). The output of the algorithm is an ordered list *S\_StructureList* containing syntactic structures of sentences in the new paragraph.

At all stages, the algorithm considers three consecutive predicates ( $Pas_{i-1}$ ,  $Pas_i$ ,  $Pas_{i+1}$ ) in *AS\_List*. The algorithm compares the priority between two pairs ( $Pas_{i-1} - Pas_i$ )

**Table 4** The syntactic structure of the new reduced Vietnamese sentence according to relationship type (iv)

Priority of $Pas_i$	Priority of $Pas_j$	Syntactic structure
(4)	(4)	$[Po1] + [Pas_i] + \langle iv \rangle + [Pas_j]$
(1)	(1)	$[Po1] + [Pas_i] + \langle iv \rangle + [Pas_j]$
(3)	(3)	$[Po1] + [Pas_i] + [Po2] + \langle iv \rangle + [Pas_j] + [Po3]$ $[Po1] + [Pas_i] + [Po2] + \langle iv \rangle + \text{"is"} + [Pas_j] + \text{"by"} + [Po3]$ $[Po1] + [Pas_i] + [Po2] + \langle iv \rangle + [Po2] + [Pas_j] + [Po3]$ $[Po1] + [Pas_i] + [Po2] + \langle iv \rangle + [Po2] + \text{"is"} + [Pas_j] + \text{"by"} + [Po3]$ $[Po1] + [Pas_i] + \langle iv \rangle + [Pas_j] + [Po2]$ $[Po1] + [Pas_i] + \langle iv \rangle + \text{"is"} + [Pas_j] + \text{"by"} + [Po2]$

**Table 5** The priority of ( $Pas_{i-1} - Pas_i$ ) is higher than the priority of ( $Pas_i - Pas_{i+1}$ )

Case	Priority of $Pas_{i-1}$	Priority of $Pas_i$	Priority of $Pas_{i+1}$
1	(X) = (1)	(Y)	(Z) > (X)
2	(X) = (2)	(Y)	(Z) > (X)
3	(X) = (3)	(Y)	(Z) > (X)

and ( $Pas_i - Pas_{i+1}$ ) and generates the syntactic structure of the new Vietnamese sentence for the pair having higher priority. With the remaining predicate, the algorithm performs one of two ways: (i) construct the syntactic structure according to this predicate—is the structure of one sentence in the original paragraph; or (ii) consider this predicate with two next predicates in *AS\_List*.

Based on classifying relationship types in Table 1, we determine priority cases between two pairs ( $Pas_{i-1} - Pas_i$ ) and ( $Pas_i - Pas_{i+1}$ ) as follows [in which (X), (Y), (Z), respectively, indicate the priority of  $Pas_{i-1}$ ,  $Pas_i$ ,  $Pas_{i+1}$ ]:

- The priority of ( $Pas_{i-1} - Pas_i$ ) is higher than the priority of ( $Pas_i - Pas_{i+1}$ ) (Table 5)
- Two priorities are equal (Table 7)

**Table 6** The priority of ( $Pas_{i-1} - Pas_i$ ) is lower than the priority of ( $Pas_i - Pas_{i+1}$ )

Case	Priority of $Pas_{i-1}$	Priority of $Pas_i$	Priority of $Pas_{i+1}$
1	(X) = (4)	(Y)	(Z) < (X)
2	(X) = (3)	(Y)	(Z) < (X)
3	(X) = (2)	(Y)	(Z) < (X)

- The priority of  $(Pas_{i-1}-Pas_i)$  is lower than the priority of  $(Pas_i-Pas_{i+1})$  (Table 6).

The algorithm for generating  $S\_StructureList$  concretely as follows:

---

**Algorithm 2:** Generate  $S\_StructureList$ .

---

**Input:**  $O\_List$  = List of predicates representing instances;  $AS\_List$  = List of predicates representing actions or states.

**Output:**  $S\_StructureList$  = List of syntactic structures of sentences in the new paragraph.

---

```

i = 2; n = |AS_List|;
While i < n Do
  Consider  $P_{i-1}, P_i, P_{i+1} \in AS\_List$ ;
  C_IAP_1 = check_inter-sentential_anaphoric_pronoun( $P_{i-1}, P_i$ );
  C_IAP_2 = check_inter-sentential_anaphoric_pronoun( $P_i, P_{i+1}$ );
  If ((C_IAP_1 == TRUE) and (C_IAP_2 == TRUE)) Then
    If (level_priority( $P_{i-1}, P_i$ ) >= level_priority( $P_i, P_{i+1}$ )) Then
      new_structure = summarize( $P_i, P_{i+1}$ );
      Put new_structure into S_StructureList;
      i = i + 2;
    Else
      new_structure = re_create( $P_{i-1}$ );
      Put new_structure into S_StructureList;
      i = i + 1;
    End
  Else If (C_IAP_1 == TRUE) Then
    new_structure = summarize( $P_i, P_{i+1}$ );
    Put new_structure into S_StructureList;
    i = i + 2;
  Else If (C_IAP_2 == TRUE) Then
    new_structure = re_create( $P_{i-1}$ );
    Put new_structure into S_StructureList;
    i = i + 1;
  Else
    new_structure = re_create( $P_{i-1}$ );
    Put new_structure into S_StructureList;
    new_structure = re_create( $P_i$ );
    Put new_structure into S_StructureList;
    i = i + 2;
  End If
End While
If (i == n+1) Then
  new_structure = re_create( $P_{i-1}$ );
  Put new_structure into S_StructureList;
Else If (i == n) Then
  C_IAP = check_inter-sentential_anaphoric_pronoun( $P_{i-1}, P_i$ );
  If (C_IAP == TRUE) Then
    new_structure = summarize( $P_{i-1}, P_i$ );
    Put new_structure into S_StructureList;
  Else
    new_structure = re_create( $P_{i-1}$ );
    Put new_structure into S_StructureList;
    new_structure = re_create( $P_i$ );
    Put new_structure into S_StructureList;
  End If
End If

```

---

In Algorithm 2, there are three important functions:

- Function `check_inter-sentential_anaphoric_pronoun( $P_x, P_y$ )` is performed to examine the inter-sentential anaphoric pronoun relationship between two sentences. This function returns TRUE if there is one in four cases:
  - Component  $S\_Index$  in  $P_x$  takes the value which is identical with the value of component  $S\_Index$  in  $P_y$ .
  - Component  $S\_Index$  in  $P_x$  takes the value which is identical with the value of component  $O\_Index$  in  $P_y$ .
- Component  $O\_Index$  in  $P_x$  takes the value which is identical with the value of component  $S\_Index$  in  $P_y$ .
- Component  $O\_Index$  in  $P_x$  takes the value which is identical with the value of component  $O\_Index$  in  $P_y$ .



- Function  $\text{summarize}(P_x, P_y)$  generates the syntactic structure of the new Vietnamese sentence for pair of predicates  $P_x, P_y$ .
- Function  $\text{re\_create}(P_x)$  constructs the syntactic structure according to predicate  $P_x$ .

To complete the new reduced Vietnamese paragraph, we replace syntactic structures by appropriate lexicon set with the following general algorithm:

**Algorithm 3:** Complete the new Vietnamese paragraph.

**Input:**  $S\_StructureList$  = List of syntactic structures of sentences in the new paragraph.

**Output:** The complete new Vietnamese paragraph.

```

For structure(i) in  $S\_StructureList$  Do
  For element(j) in structure(i) Do
    If element(j) is predicate Then
      Replace element(j) by its
      represented lexicon;
    Else If element(j) is relationship
    type Then
      Replace element(j) by appropriate
      lexicon set in Vietnamese;
    End If
  End For
End For

```

In this research, we use the Vietnamese lexicon set suitable for each relationship type in Table 1 as follows:

- Relationship type (i): “*đề*” (English: for).
- Relationship type (ii): “*vì*” (English: because/because of).
- Relationship type (iii): “*nên*” (English: so).
- Relationship type (iv): “*và*” (English: and).

Apply Algorithm 2 for  $O\_List$  and  $AS\_List$  containing predicates of the logical expression in Sect. 1 (described in Sect. 4.1) as follows:

- $n = |AS\_List| \rightarrow 4$ ;
- $i = 2 < n$ ;
- Consider three predicates:
  - $P_1 = \text{vui\_vẻ} (x, \text{state}, \text{status})$
  - $P_2 = \text{học\_cúng} (x, y, \text{action}, \text{transitive})$
  - $P_3 = \text{khoái\_chí} (y, \text{state}, \text{status})$ .
- Check inter-sentential anaphoric pronoun:
  - $C\_IAP\_1 = \text{TRUE}$  because component  $S\_Index$  in  $P_1$  takes the value which is identical with the value of component  $S\_Index$  in  $P_2$ .

**Table 7** Two priorities are equal

Case	Priority of $Pas_{i-1}$	Priority of $Pas_i$	Priority of $Pas_{i+1}$
1	(X)	(Y)	(Z) = (X)

- $C\_IAP\_2 = \text{TRUE}$  because component  $O\_Index$  in  $P_2$  takes the value which is identical with the value of component  $S\_Index$  in  $P_3$ .
- According to Table 7:  $\text{level\_priority}(P_1, P_2) = \text{level\_priority}(P_2, P_3)$ .
- $\text{new\_structure} = \text{summarize}(P_1, P_2)$ .
  - According to Table 8:  $\text{new\_structure} = [\text{lan} (x)] + [\text{vui\_vẻ} (v)] + <ii> + [\text{học\_cúng} (x, y)] + [\text{con\_trai} (y)]$ .
- Put  $\text{new\_structure}$  into  $S\_StructureList$ ;
- $i = i + 2 \rightarrow 4$ ;
- $i = 4 = n$ ;
- Consider two predicates:
  - $P_3 = \text{khoái\_chí} (y, \text{state}, \text{status})$
  - $P_4 = \text{được} (y, z, \text{action}, \text{transitive})$ .
- Check inter-sentential anaphoric pronoun:
  - $C\_IAP = \text{TRUE}$  because component  $S\_Index$  in  $P_3$  takes the value which is identical with the value of component  $S\_Index$  in  $P_4$ .
- $\text{new\_structure} = \text{summarize}(P_3, P_4)$ .
  - According to Table 8:  $\text{structure} = [\text{con\_trai} (y)] + [\text{khoái\_chí} (y)] + <ii> + [\text{đủ'ợ'c} (y, z)] + [\text{điểm\_cao} (z)]$ .
- Put  $\text{new\_structure}$  into  $S\_StructureList$ ;
- Apply Algorithm 3, we obtain the result is the new reduced Vietnamese paragraph:

“*Lan vui vẻ vì học cùng con trai. Con trai khoái chí vì được điểm cao.*”

(English: “Lan is happy because of studying with the son. The son is overjoyed because of taking high mark.”).

## 5 Experiment and analysis

To perform the experiment and evaluate the success rate, we establish two criteria with concrete marks:

**Table 8** The syntactic structure of the new reduced Vietnamese sentence according to relationship type (ii)

Priority of $Pas_i$	Priority of $Pas_j$	Syntactic structure
(1)	(4)	$[Po1] + [Pas_i] + \langle ii \rangle + [Pas_j]$
(1)	(2)	$[Po1] + [Pas_i] + \langle ii \rangle + [Pas_j]$
(1)	(3)	$[Po1] + [Pas_i] + \langle ii \rangle + [Pas_j] + [Po2]$
		$[Po1] + [Pas_i] + \langle ii \rangle + "is"[Pas_j] + "by" + [Po2]$
(2)	(4)	$[Po1] + [Pas_i] + \langle ii \rangle + [Pas_j]$
(3)	(4)	$[Po1] + [Pas_i] + [Po2] + \langle ii \rangle + [Pas_j]$
		$[Po1] + [Pas_i] + [Po2] + \langle ii \rangle + [Po2] + [Pas_j]$

**Table 9** The testing results

Case	Number	/No1	/No2	/No3
The number (No1) of logical expressions	500			
The number (No2) of new reduced paragraphs	428	0.856		
The number (No3) of new reduced paragraphs satisfying mark 1 of the first criteria	397	0.794	0.928	
The number (No4) of new reduced paragraphs satisfying mark 1 of the second criteria	136		0.318	0.343
The number (No5) of new reduced paragraphs satisfying mark 2 of the second criteria	234		0.547	0.589

- The first criterion is the semantic correctness with two marks: 1—correctness; 0—not correctness. This criterion is evaluated based on manually considering that the new reduced paragraph correctly summarizes the meaning of the original paragraph or not.
- The second criteria are the universality in Vietnamese with three marks: 2—universality if every sentences in the new reduced paragraph have the universality; 1—acceptable if there is one sentence in the new reduced paragraph which does not totally have the universality; 0—do not have the universality when there are two or more sentences which do not have the universality.

Based on these two criteria, we built the testing data set consisting of Vietnamese paragraphs according to the rule with the following points:

- Each paragraph is composed of 3–5 Vietnamese sentences having simple structure.
- If there are three or more consecutive sentences in which each pair of sentences does not have the inter-sentential anaphoric pronoun relationship, then the paragraph is fairly trivial to summary. Therefore, we require at least at the second and the forth sentence there are the occurrences of the anaphoric pronouns.

With the above rule, we collected 500 Vietnamese paragraphs and constructed 500 logical expressions for testing. The results are presented in Table 9 as follows:

Analyzing the results in Table 9, we see that

- With the central is Algorithm 2, the solution showed the effectiveness in generating new reduced paragraphs which satisfy the above criteria.
- There are some limitations with causes:
  - Because there is no additional factor showing the context about space and time in which the fact happened, therefore, we determined the inter-sentential relationships based on the assumption in Sect. 2.2. This leads to the generated paragraph may not have totally semantic correctness or universality in a reality context.
  - In some logical expressions, there are predicates representing actions or states which have component  $S\_Index$  or  $O\_Index$  taking the value which does not indicate the correct object. This leads to cannot generate or the new generated paragraph does not have the semantic correctness.

These limitations will become our main objectives in the next researches.

## 6 Discussion and conclusion

In this paper, we presented the algorithm of generating the new reduced Vietnamese paragraph from the logical expression of DRS encoding the semantics of the source paragraph. This algorithm computes the verbal relationships between related sentences, based on the proposed assumptions. The experiment shows that the quality of new summarization paragraphs is enhanced and considerably conformable to Vietnamese native speakers.

We also pointed out some limitations of this solution. These limitations will be studied and overcome in future researches, focused on following main points: consider paragraphs having more complex structure, and try to find other assumptions which are more universal in Vietnamese.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Blackburn, P., Bos, J.: Representation and Inference for Natural Language—Volume II: Working with Discourse Representation Structures. Department of Computational Linguistics, University of Saarland, Saarbrücken (1999)
- Barzilay, R., et al.: Information fusion in the context of multi-document summarization. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 550–557 (1999)
- Barzilay, R., McKeown, K.R.: Sentence fusion for multidocument news summarization. *Comput. Linguist.* **31**, 297–328 (2005)
- Cao, H.X.: Tiếng Việt: Sơ thảo ngữ pháp chức năng [Vietnamese: Brief of Functional Grammar]. Nhà xuất bản giáo dục [Education Publisher] (2006)
- Covington M.A., Schmitz, N.: An Implementation of Discourse Representation Theory. ACMC Research Report Number: 01-0023. Advanced Computational Methods Center, The University of Georgia, Athens (1989)
- Covington, M.A., Nute, D., Schmitz, N., Goodman, D.: From English to Prolog Via Discourse Representation Theory. ACMC Research Report Number: 01-0024. Advanced Computational Methods Center, University of Georgia, Athens (1988)
- Das, D., Martins, A.F.T.: A Survey on Automatic Text Summarization. Language Technologies Institute, Carnegie Mellon University, Pittsburgh (2007)
- Genest, P.E., Lapalme, G.: Fully abstractive approach to guided summarization. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers, vol. 2, pp. 354–358 (2012)
- Genest, P.E., Lapalme, G.: Framework for abstractive summarization using text-to-text generation. In: Proceedings of the Workshop on Monolingual Text-To-Text Generation, pp. 64–73 (2011)
- Greenbacker, C.F.: Towards a framework for abstractive summarization of multimodal documents. *ACL HLT* **2011**, 75 (2011)
- Halliday, M.A.K., Matthiessen, C.M.I.M.: An Introduction to Functional Grammar, 3rd edn. Hodder Arnold, London (2004)
- Harabagiu, S.M., Lacatusu, F.: Generating single and multi-document summaries with gistexter. In: Document Understanding Conferences (2002)
- Jezek, K., Steinberger, J.: Automatic text summarization. In: Snasel, V. (ed.): *Znalosti 2008*, ISBN 978-80-227-2827-0, FIIT STU Bratislava, Ustav Informatiky a softveroveho inzinierstva, pp. 1–12 (2008)
- Jones, K.S.: Automatic summarizing: factors and directions. In: Mani, I., Marbury, M. (eds.): *Advances in Automatic Text Summarization*. MIT Press, Cambridge (1999)
- Jones, K.S.: Automatic Summarising: A Review and Discussion of the State of the Art. Technical Report 679. Computer Laboratory, University of Cambridge, Cambridge (2007)
- Kamp, H.: A theory of truth and semantic representation. In: Groenendijk, J., Janssen, T.M.V., Stokhof, M. (eds.): *Formal Methods in the Study of Language, Part 1. Mathematical Centre Tracts. Mathematical Centre Tracts*, pp. 277–322 (1981)
- Kasture, N.R., Yargal, N., Singh, N.N., Kulkarni, N., Mathur, V.: A survey on methods of abstractive text summarization. *Int. J. Res. Merg. Sci. Technol.* **1**(6), 53–57 (2014)
- Khan, A., Salim, N.: A review on abstractive summarization methods. *J. Theor. Appl. Inf. Technol.* **59**(1), 64–72 (2014)
- Lee, C.S., et al.: A fuzzy ontology and its application to news summarization. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **35**, 859–880 (2005)
- Lloret, E.: Text summarization: an overview. In: Paper Supported by the Spanish Government Under the Project TEXT-MESS (TIN2006-15265-C06-01) (2008)
- Mani, I., Maybury, M.T.: *Advances in Automatic Text Summarization*. MIT Press, Cambridge (1999)
- Moawad, I.F., Aref, M.: Semantic graph reduction approach for abstractive text summarization. In: Seventh International Conference on Computer Engineering and Systems (ICCES), pp. 132–138 (2012)
- Reiter, E., Dale, R.: *Building Natural Language Generation System*. Cambridge University Press, Cambridge (1997)
- Reiter, E., Dale, R.: Building applied natural language generation systems. *Nat. Lang. Eng.* **3**(1), 57–87 (1997)
- Saranyamol, C.S., Sindhu, L.: A survey on automatic text summarization. *Int. J. Comput. Sci. Inf. Technol.* **5**(6), 7889–7893 (2014)
- Tanaka, H., et al.: Syntax-driven sentence revision for broadcast news summarization. In: Proceedings of the 2009 Workshop on Language Generation and Summarisation, pp. 39–47 (2009)
- Tran, T., Nguyen, D.T.: Merging two Vietnamese sentences related by inter-sentential anaphoric pronouns for summarizing. In: Proceedings of the 1st NAFOSTED Conference on Information and Computer Science (NICS'14), Hanoi, pp. 371–381 (2014)
- Tran, T., Nguyen, D.T.: Improving techniques for summarizing the meaning of two Vietnamese sentences by adding a meaningful relationship between two actions. In: Proceedings of the 16th ACM International Conference on Information Integration and Web-based Applications and Services (iiWAS'14), Hanoi, pp. 484–488 (2014)
- Tran, T., Nguyen, D.T.: Enhancement of sentence-generation based summarization method by modelling inter-sentential consequent-relationships. In: Proceedings of the 16th ACM International Conference on Information Integration and Web-Based Applications and Services (iiWAS'14), Hanoi, pp. 302–309 (2014)
- Tran, T., Nguyen, D.T.: Specification model of paragraph summarization by verbal relationships: objective, cause, consequence, concurrence. In: Proceedings of the 2nd IEEE International Conference on Artificial Intelligence, Modelling and Simulation (AIMS'14), Madrid, pp. 205–210 (2014)
- Tran, T., Nguyen, D.T.: Semantic predicative analysis for resolving some cases of ambiguous referents of pronoun “Nó” in summarizing meaning of two Vietnamese sentences. In: Proceedings of the 17th UKSIM-AMSS International Conference on Modelling and Simulation (UKSIM'15), Cambridge, pp. 340–345 (2015)
- Tran, T., Nguyen, D.T.: Combined method of analyzing anaphoric pronouns and inter-sentential relationships between transitive verbs for enhancing pairs of sentences summarization. In: Silhavy, R. (eds.): *Proceedings of the 4th Computer Science On-line Conference (CSOC'15)—Vol 1: Artificial Intelligence Perspectives and Applications. Advances in Intelligent Systems and Computing*, vol. 347, pp. 67–77. Faculty of Applied Informatics, Tomas Bata University in Zlin, Czech Republic (2015)

33. Tran, T., Nguyen, D.T.: Modelling consequence relationships between two action, state or process Vietnamese sentences for improving the quality of new meaning-summarizing sentence. *Int. J. Pervasive Comput. Commun.* **11**(2), 169–190 (2015). (Emerald Group Publishing Limited. ISBN 1742-7371)
34. Zettlemoyer, L.S., Collins, M.: Learning to map sentences to logical form: structured classification with probabilistic categorial grammars. In: *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI'05)*, pp. 658–666 (2005)
35. Zettlemoyer, L.S., Collins, M.: Online learning of relaxed CCG grammars for parsing to logical form. In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL'07)*, pp. 678–687 (2007)
39. Le, H.T., Sam, R.C., Nguyen, P.T.: Extracting phrases in Vietnamese document for summary generation. In: *Proceedings of International Conference on Asian Language Processing (IALP)*, Harbin, pp. 207–210 (2010)
40. Tran, T., Nguyen, D.T.: Improve effectiveness resolving some inter-sentential anaphoric pronouns indicating human objects in Vietnamese paragraphs using finding heuristics with priority. In: *Proceedings of the 10th IEEE RIVF International Conference on Computing and Communication Technologies–Research, Innovation, and Vision for the Future (RIVF'13)*, Hanoi, pp. 109–114 (2013)
41. Tran, T., Nguyen, D.T.: A solution for resolving inter-sentential anaphoric pronouns for Vietnamese paragraphs composing two single sentences. In: *Proceedings of the 5th International Conference of Soft Computing and Pattern Recognition (SoCPaR'13)*, Hanoi, pp. 172–177 (2013)
42. Tran, T., Nguyen, D.T.: Implementation of a discourse representation based approach for summarization of Vietnamese text paragraphs. In: *Proceedings of the 3rd Asian Conference on Information Systems (ACIS'14)*, Nha Trang, pp. 275–282 (2014)

## Further Readings

36. Covington, M.A.: GULP 4: An Extension of Prolog for Unification Based Grammar. Research Report Number: AI-1994-06. Artificial Intelligence Center, The University of Georgia, USA (2007)
37. Gupta, V., Lehal, G.S.: A survey of text summarization extractive techniques. *J. Emerg. Technol. Web Intel.* **2**(3), 258–268 (2010)
38. Le, H.T., Le, T.M.: An approach to abstractive text summarization. In: *Proceedings of the 5th International Conference of Soft Computing and Pattern Recognition (SoCPaR'13)*, Hanoi, pp. 372–377 (2013)